A Customer Focused Approach to Architecting for Software Reliability

Mario Garzia
Björn Levidow
Windows Reliability
Microsoft Corporation

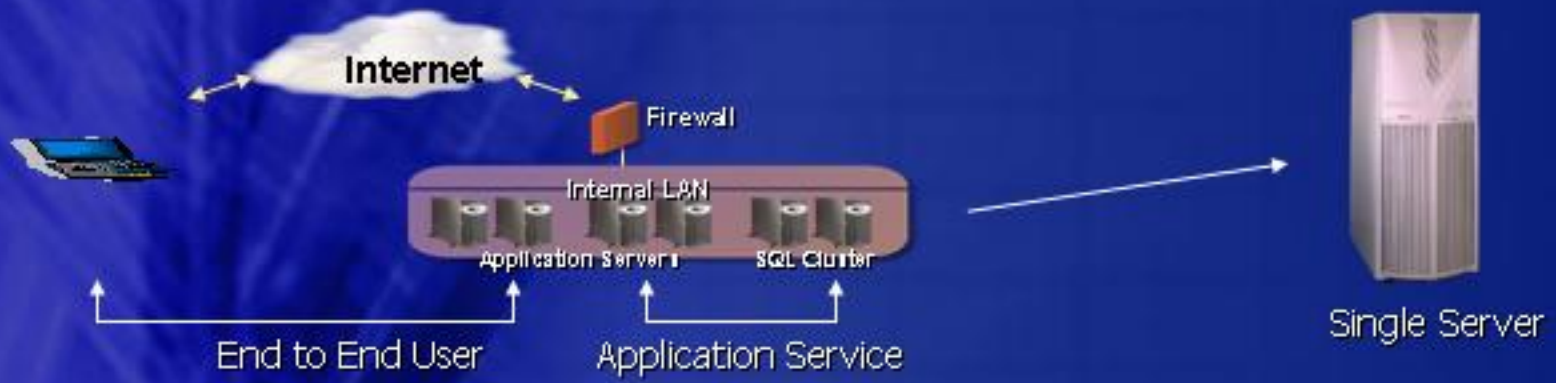
Agenda

- Approach to Reliability Engineering
- Reliability Data Collection and Analysis
  - Demo: Microsoft Reliability Service
- Windows Reliability Improvement
- Windows Server Reliability Features
  - Demo: Hotpatching
  - Demo: Software Tracing
- Topics for Further Investigation
- Links to Resources

System And Service Reliability

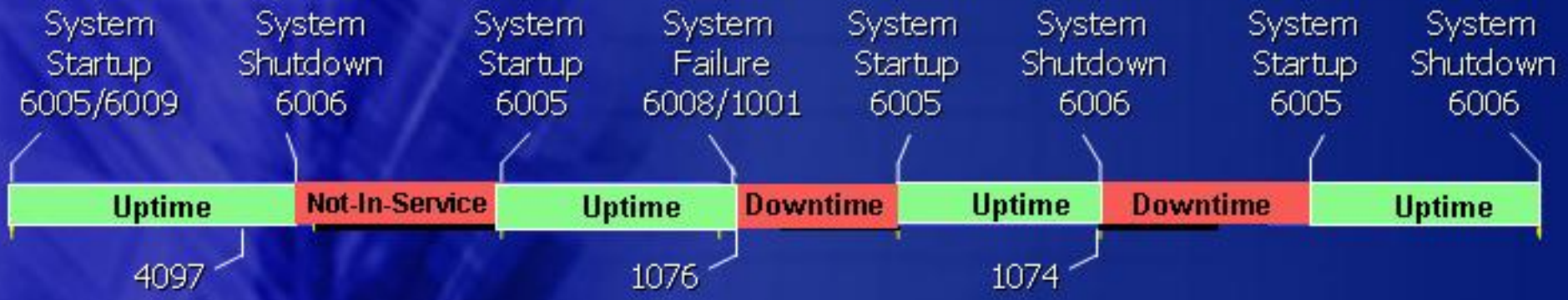
- Customers want reliable systems and services
  - Systems and services work the way they should and are easy to install, manage and use
  - Rock-solid and bug free basic functionality
  - Compatibility across releases and applications
  - Dependable support and documentation when problems arise
  - Components that are resilient to faults
- Corporations need to provide high levels of service availability at the lowest possible cost
  - Revenue and customer satisfaction depend on achieving high availability
  - Low cost is critical to remain competitive
- Engineering highly reliable products and services requires focus on reliability throughout the lifecycle

Reliability & Availability Perspectives



- Reliability – measure of the time to a system/service impacting event
- Availability – percentage of time that a system or service is performing its intended function
- Three commonly used perspectives
  - Single Server Reliability/Availability – refers to the reliability/availability of an individual single server system
  - Application Service Reliability/Availability – is the reliability/availability of all systems needed to provide the service
  - End to End User Reliability/Availability – refers to the reliability/availability of the application service, Internet connectivity and end user client
- Additional breakdown is possible in terms of Hardware, OS and Application software

Tracking Windows Availability States



- **Three system states considered**
  - Available - Includes logged off and standby (uptime)
  - Unavailable - Includes OS crash and app installations (downtime)
  - Not-In-Service (NIS) - Customer turns off computer (not in use)
- **System/Application Event Log**
  - Startup event (6005), Clean shutdown event (6006), Dirty shutdown event (6008)
  - Additional events are used to identify system characteristics and to carry out root cause analysis
    - E.g., System version event (6009), Service pack installation (4353), Save dump (1001), Reboot Annotation Events (1073 to 1076), Dr Watson (4097)
- **OS Uptime = time stamp 6006/8 – time stamp 6005**
- **OS Downtime = time stamp 6005 – time stamp 6006/8**
- **Similar uptime/downtime metrics defined for application events**

Reliability And Availability Metrics

- OS Reliability Metrics
  - Mean Time To a Reboot → MTTReboot = Σ uptime / # Reboots
  - Mean Time To a Failure → MTTFailure = Σ uptime / # Failures
    - Based on 6008 events
  - Mean Time to a Blue Screen → MTTBS = Σ uptime / # Crashes
    - Based on 1001 events
  - Mean Time to Restore → MTTRestore = Σ downtime / # Reboots
- Application Reliability Metrics
  - Mean Time To Application Stop → MTTAS
    - Based on, for example, SQL start event 17162
  - Mean Time To Application Restore → MTTARestore
- Availability Metrics
  - Single Server Availability = Σ uptime / (Σ uptime + Σ downtime)
  - Single Server and Clustered Server Availability
  - Application Availability ≤ OS Availability

Windows Reliability Testing

- Standard Software Testing
  - Automated and manual tests
  - Unit, component, integration and long-haul stress tests
  - Compatibility, fault injection and H/W tests
- Customer Deployment Scenario Testing
  - Customer environments replicated and tested during development process
- LongHaul and Deployment Server Reliability Tracking
  - Collection of comprehensive data on hundreds of servers focusing on robustness and reliability
  - Quantifiable statistics on server product line during the development process
  - Historical data from prior versions of the OS used as benchmark

Windows Server Reliability Ship Criteria

- Product reliability readiness assessment in a production environment prior to shipping software
- Reliability assessment of production servers on major escrow builds
  - WS3 assessment of OTG servers
  - Assessment of future versions to include internal and external production servers

- Reliability target established based on expected improvement over prior OS version
  - Criteria used for product sign-off
  - Bounds capture variability from small runtime/server samples
  - Observed availability growth throughout development cycle
  - Significantly higher measured availability with Windows Server 2003

Ongoing Customer Assessment

- Production server measurement program started in earnest with NT4 and provides detailed information for improving each subsequent version of the OS
  - NT4 measurements leading to Windows 2000 improvements
    - Measured 4,600 NT4 SP3/SP4 servers at 15 sites
    - All 24x7, professionally run large sites
  - Windows 2000 measurements
    - Measured over 10,000 Windows 2000 production servers at more than 30 sites
    - Sites measured include Financial, High Tech and other companies running mission critical applications
  - Windows Server 2003 measurements
    - Pre-release reliability assessment of JDP customers
    - Ship criteria based on meeting reliability objectives
    - Windows Server 2003 measurements will be possible at thousands of customer sites
- Client corporate and consumer measurements
- Customer reliability surveys for client and server

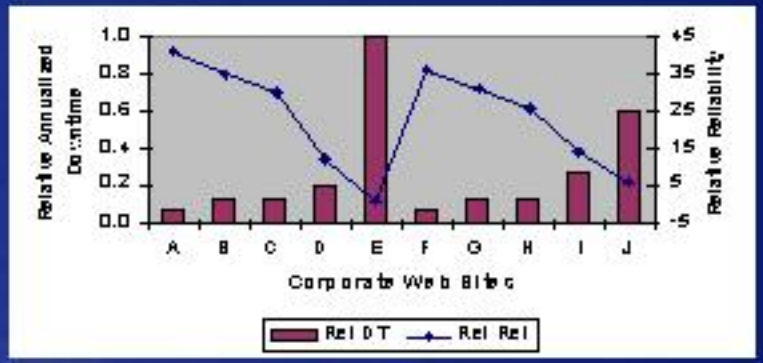
Assessing Reliability

- **Analyzing reliability data**
  - Lack of widely accepted analysis and definitions standards making data comparisons difficult
  - Insufficient analytical techniques for analyzing software reliability
  - Lack of automated tools for data analysis
- **Many variables impacting reliability results**
  - Differences in hardware, software (applications and OS), and procedures
  - External events, traffic characteristics and time periods
- **Many data analysis assumptions required**
  - System reboots considered to be independent of each other
  - Initial system instability periods, short runtimes, NIS times, non-production systems filtering
  - Client data
    - Not-In-Service time is discounted
    - Data for each PC analyzed separately
  - Adjustments made to reflect specific business practices
  - Shutdown annotations required for data interpretation
  - Server data
    - Analysis assumes 24x7 server operation
    - Data for similar systems is combined for analysis

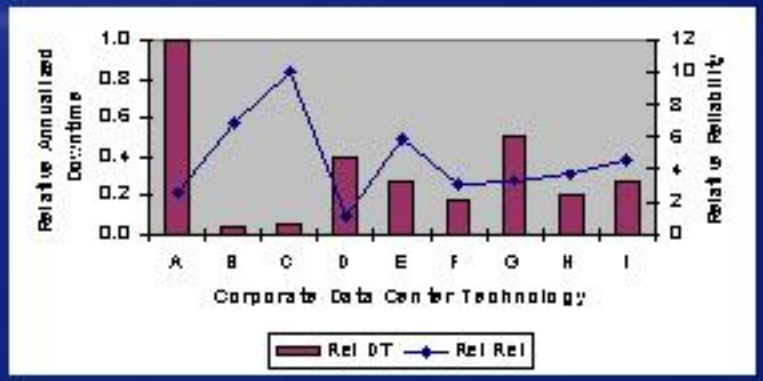
Comparing Availability Results

- We cannot talk about system or OS availability as if it were a single number, it depends on many variables
  - Reliability benchmarks needed for broad comparisons

- Same technology, similar hardware, same OS. Different data centers, applications and procedures. Very different results

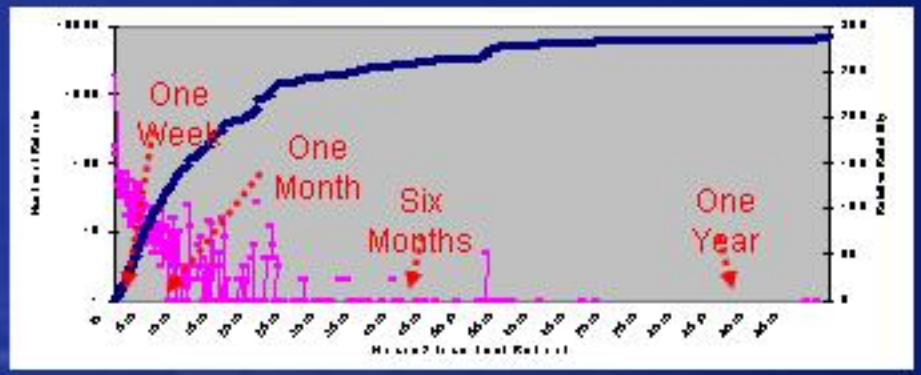


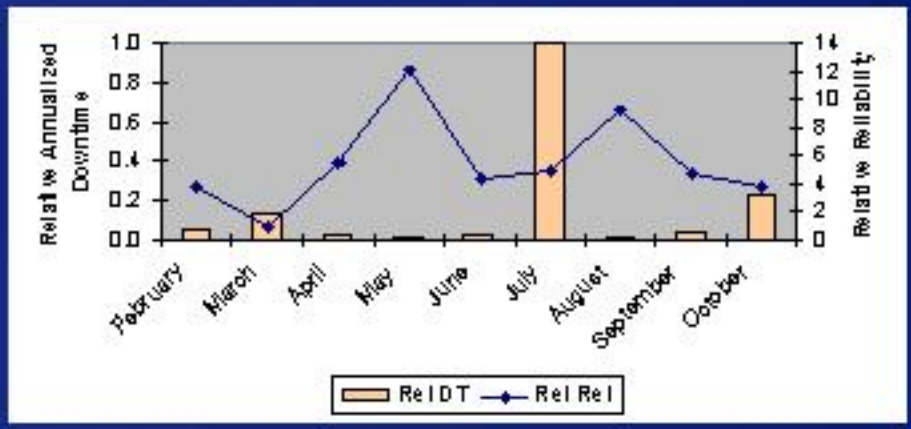
- Single data center, standard hardware, same procedures and OS. Different technologies and applications. Very different results



Measurement Period Impacts Results

- Data measurement period needs to be sufficiently long so that enough of the system behavior can be observed
- Times between system reboots exhibit great variability, from reboots that are within 15 minutes of each other to systems that remain up for over a year
- On a monthly basis, both reliability and availability can vary greatly due to system or procedural issues



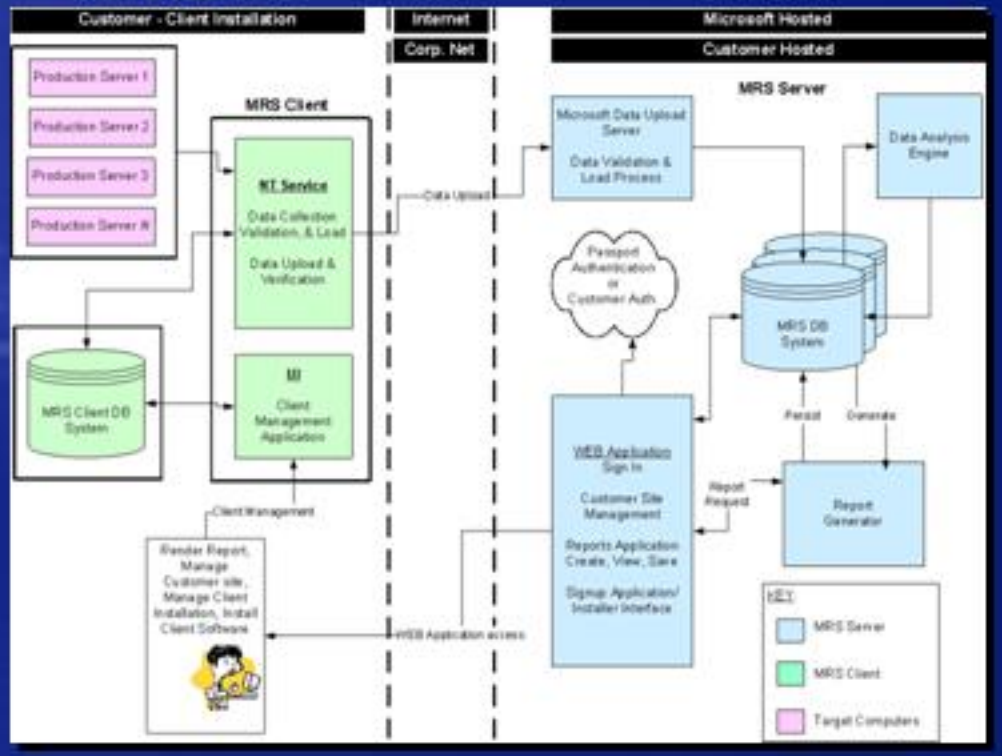


The Microsoft Reliability Service

- The reliability service gathers event log data from customer data center servers, analyzes the data and produces tailored reliability and availability reports for the customer
  - Leverages the instrumentation built into the Windows OS and major MS Applications
  - Tracks and reports on all OS and Application shutdowns
  - Provides consistent reliability measurements reporting to customers running Windows Servers
  - Provides feedback loop to the customer on areas for reliability improvement
  - Enhances our ability to improve products through a deeper and broader understanding of customer problems

Architecture

- MRS consists of Client and Reporting Site components
- MRS can be used in one of two modes
  - Customer hosted – both the Client and Reporting Site components are hosted at the customer site (customer can optionally share data with Microsoft)
  - Microsoft hosted – the MRS client is installed at the customer site but the Reporting Site component resides at Microsoft
- Architectural features:
  - MRS Client is light weight
    - Installs on single server
    - No agents on target server
    - Negligible performance impact
  - WEB based reporting
  - Incremental data uploads
  - Web based installer
  - Customer hosted option
    - Single server implementation
  - Microsoft hosted service
    - Passport authentication
    - Data partitioning



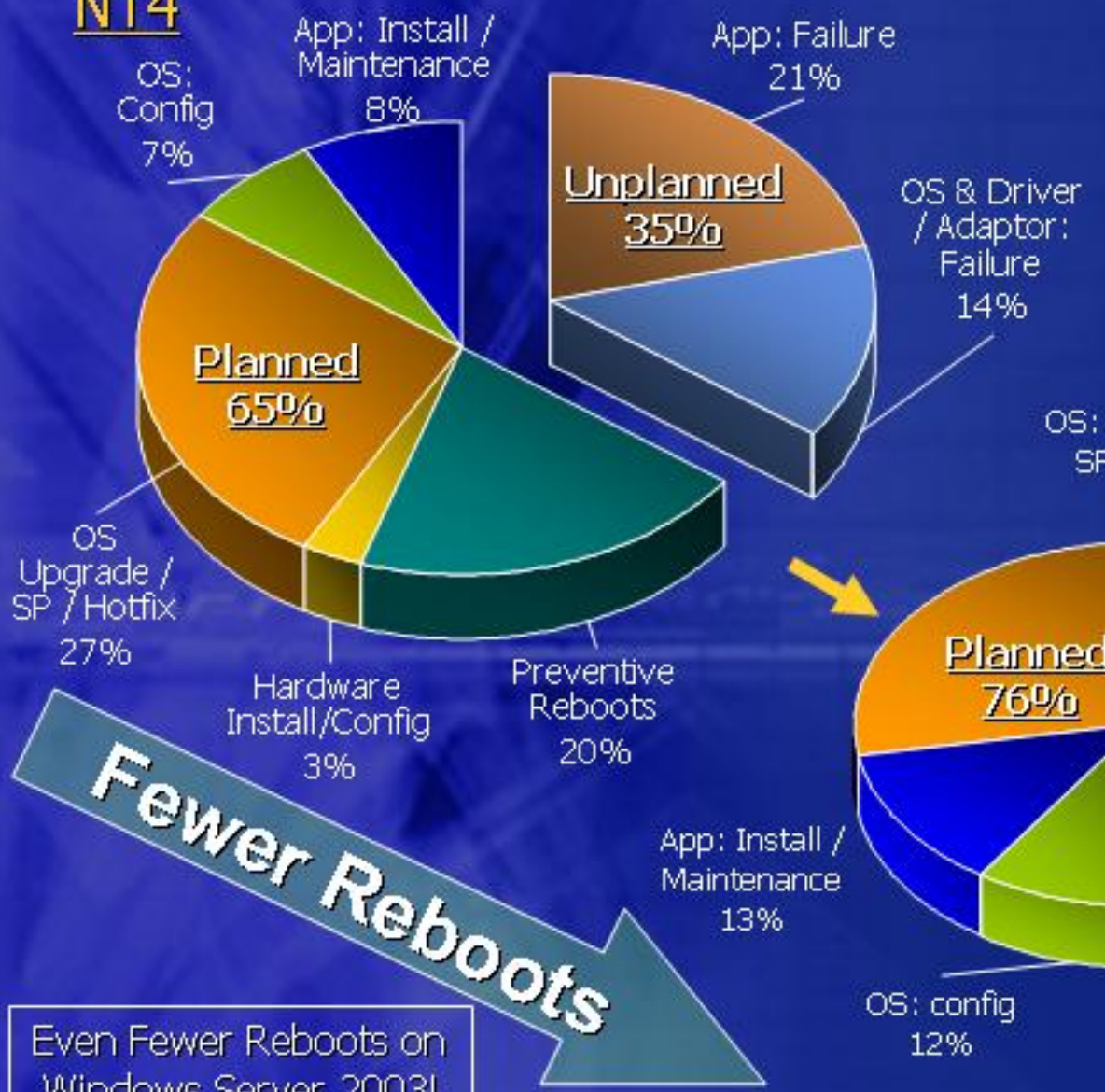
Microsoft Reliability Service

demo

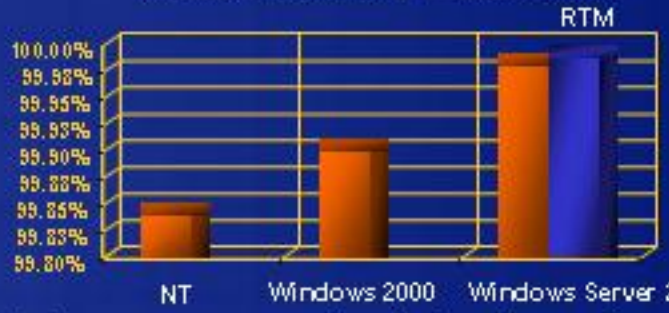
Heidi Crittenden
Dave Crocco
Windows Reliability Group

Windows Reliability Improvement

NT4



Server Availability Improvement



Data measures servers at multiple customer sites and includes all planned and unplanned shutdowns.

Windows 2000

Fewer Reboots

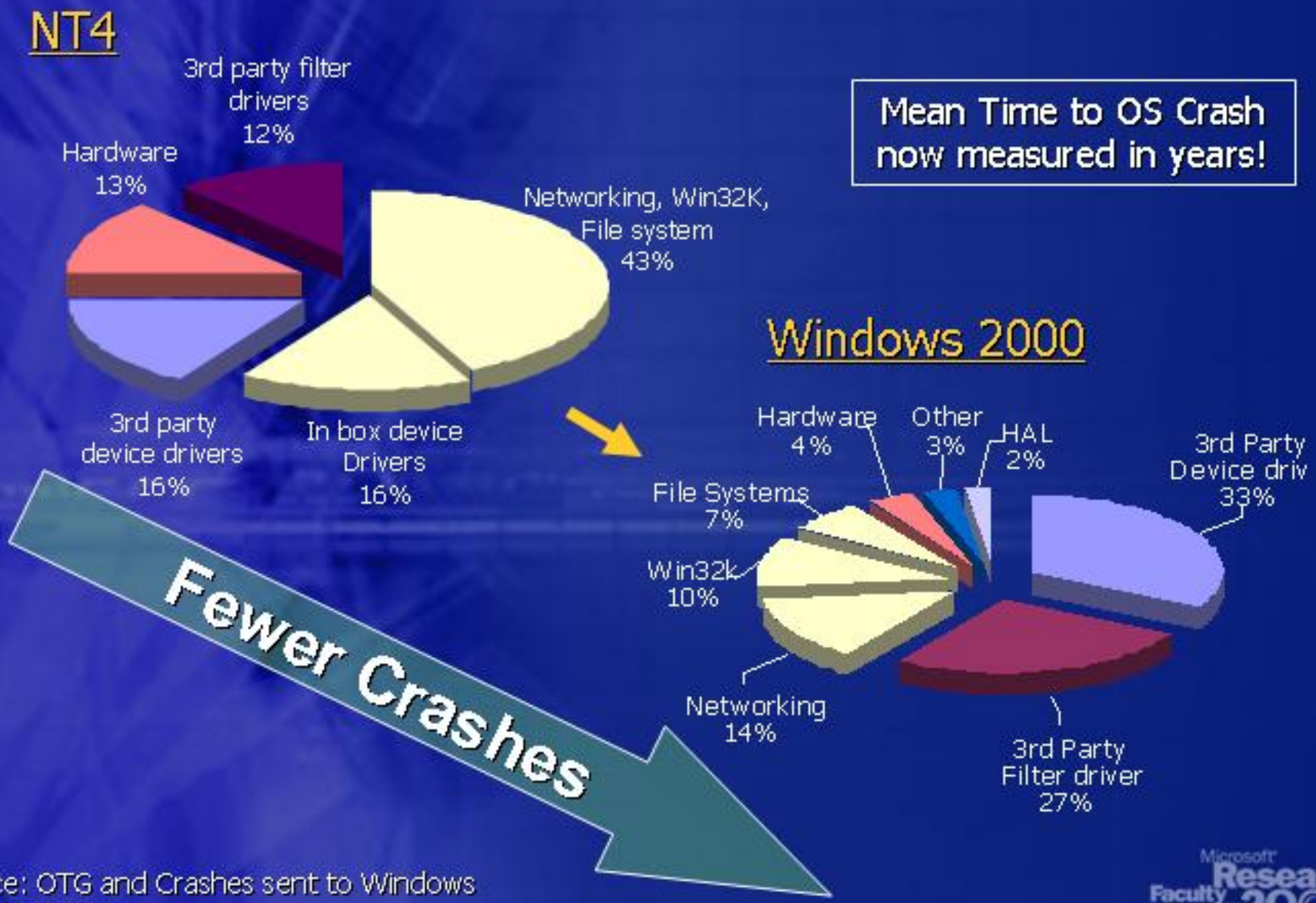
Even Fewer Reboots on Windows Server 2003!

Microsoft Research Faculty Summit 2003

Windows Error Reporting

- Provides Kernel and User mode crash and hang reporting to Microsoft
  - Broad source of feedback for product improvement
  - Automatic analysis and signature matching to known issues
  - Dedicated Microsoft resources for finding solutions to common application and system crashes in Windows and 3rd party software
  - Corporations control sending information using the corporate error reporting client
  - Available to Premier customers on Windows 2000 and all Windows XP and Windows Server 2003 customers
  - Reporting will be later expanded to cover other cases including customer feedback with necessary fixes
- Microsoft uses the data solely to improve quality
  - Identify main customer pain points across MS products
  - Share data with 3rd parties to improve ecosystem

Windows Reliability Improvement



Source: OTG and Crashes sent to Windows Error Reporting

Windows Server 2003 Built To Surpass Windows 2000 Reliability

- Many great reliability enhancing features
- High Availability Release Criteria
  - Availability goals established and tracked for each major build as criteria for shipping
- Aggressive pre-release production quality assessment
  - Hundreds of customer and internal production servers on Windows Server 2003 monitored
  - MRS and WER problem identification and resolution
- Prefix/Prefast integration
  - Desktop static analysis tools help developers find bugs before check-in
  - Simulation analysis tools in build process find more complex coding mistakes

Windows Server Reliability Features

OS Upgrade / Hotfix

- Enhanced QFE testing to remove unnecessary reboots
- QFE Chaining
  - Allows multiple QFEs to be installed with a single reboot
  - Ensures latest version of DLLs across hotfix packages
- Hotpatching
  - Automatic insertion of code into a running process allowing it to be modified without reboots or service interruptions
  - Changes must be contained in one function, or no dependencies between functions being patched
  - Works for ~30% - 40% of patches

Hotpatching

demo

Adrian Marinescu
Windows Kernel Group

Diagnostic System Instrumentation

- Driver Verifier Improvements
- Pool tagging enhancements
  - Low overhead implementation on by default
  - Allows easier diagnosis of driver memory leaks with less downtime
- Enhanced software tracing
  - Allows developers to write out debug messages to a log
  - More components instrumented natively
  - High performance: 8K/sec events < 2% CPU

Software Tracing

demo

Jeff Meng
Windows Reliability Group

Reboot Annotations

- Shutdown Event Tracker (SET)
  - Captures system generated and user entered reboot reasons
  - Captures user's intent, observed symptoms or cause
    - Collected by Reliability Service
  - Captures snapshot of the system state when user selects "unplanned" reason

Reboot

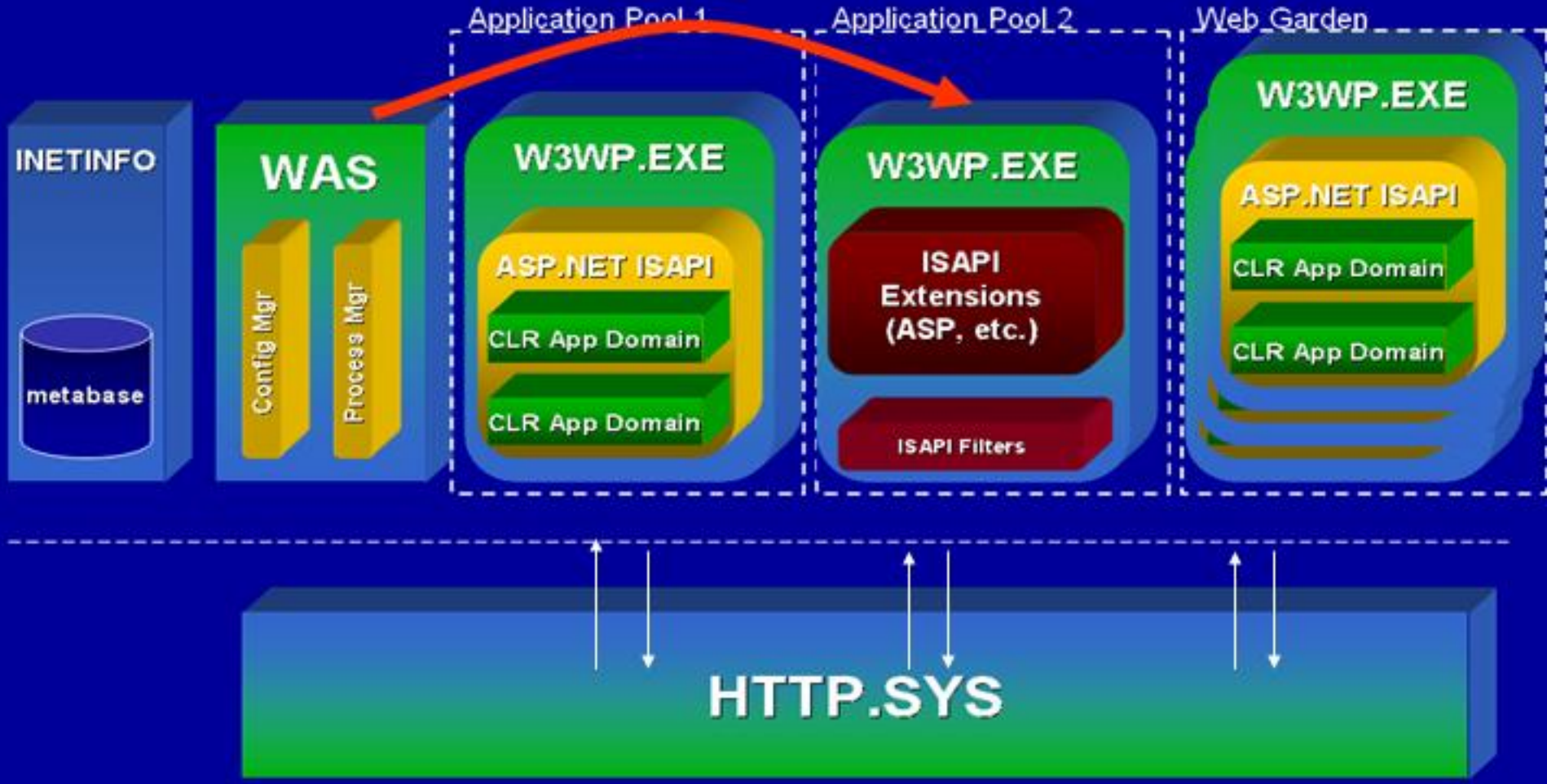
Reason is System Generated where known or User Supplied

Architectural Improvements in Windows Server 2003

- Windows Resource Management
  - Set allocation (CPU and memory) policies on applications
    - Select processes and set resource consumption limits
  - Manages CPU utilization, Process working set size and committed memory
  - Apply policies on a date/time schedule
  - Keeps, displays and export accounting records
- IIS 6 Automated Application Recycling
  - Avoid possible leak/fragmentation problems
  - Both pooled and isolated applications
  - Recycle based on various criteria
- COM+ Application Recycling
  - Recycle based on memory utilization, method calls processed, activation count or time in use
- Volume Snapshot Services

IIS 6.0 Architecture

The transition from IIS5 to IIS6



Topics For Further Investigation

- Reliability Benchmarking
  - Standard, repeatable, cross-platform measurements
- Reliability Analysis Standards
- Consistent, specific, complete, definitions of terminology
  - Confusion with meaning of "reliability" is common
- Standard methodologies for dealing with un-trusted components
- How can a large, automated community be enabled to improve customer reliability?
  - Driver quality
- Automated diagnosis and repair
  - How much automation is too much?
  - How do you get customers to accept it?

Links To Resources

- Windows Error Reporting
  - http://watson.microsoft.com/dw/1033/dcp.asp
  - https://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/prodtechnol/winxppro/proddocs/sysdm_fault_reporting_overview.asp
- Software Tracing
  - http://msdn.microsoft.com/library/default.asp?url=/library/en-us/ddtools/hh/ddtools/st_32pf.asp
- Driver Verifier
  - http://msdn.microsoft.com/library/default.asp?url=/library/en-us/ddtools/hh/ddtools/dv_9rw3.asp
- Application Verifier
  - http://msdn.microsoft.com/library/default.asp?url=/library/en-us/appverif/appverif/overview_of_application_verifier.asp

Microsoft®

Microsoft Research Faculty Summit 2003